

WHEN THE DUCK QUACKS: MULTIMODAL QUERYING WITH FLOCKMTL

A n a s D o r b a n i



**FlockMTL
Website**

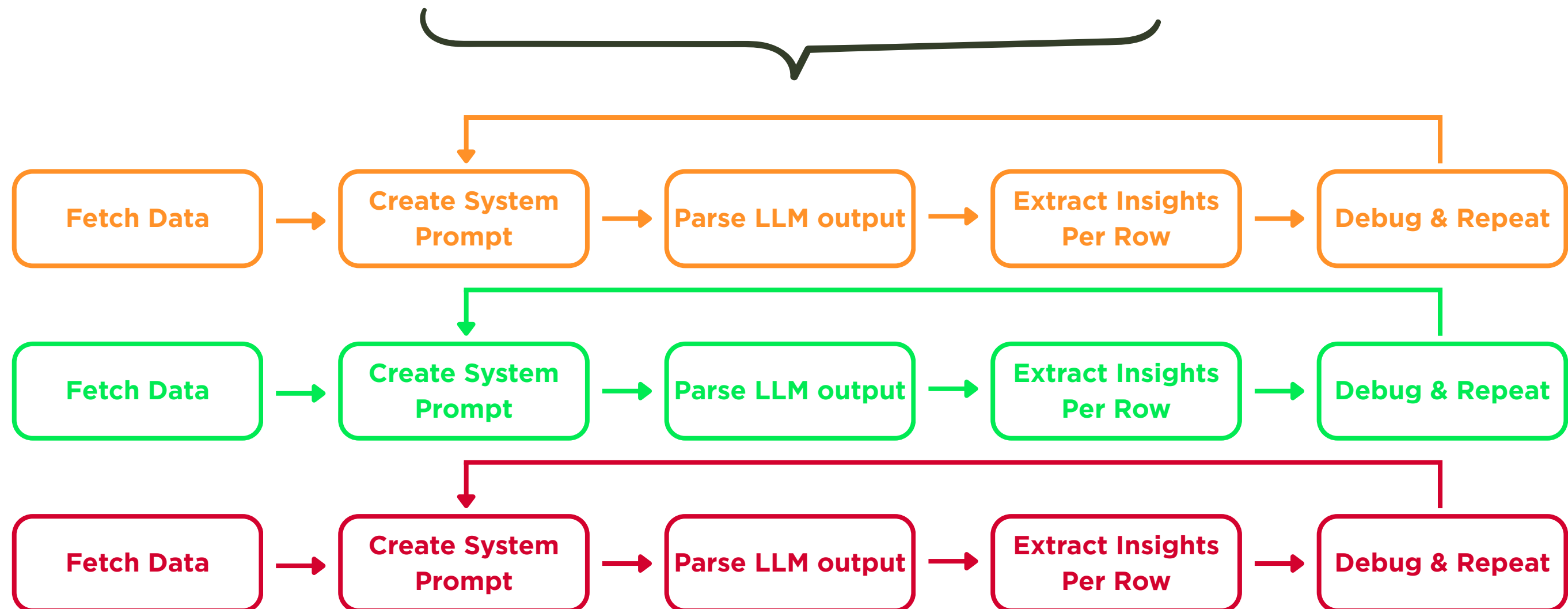
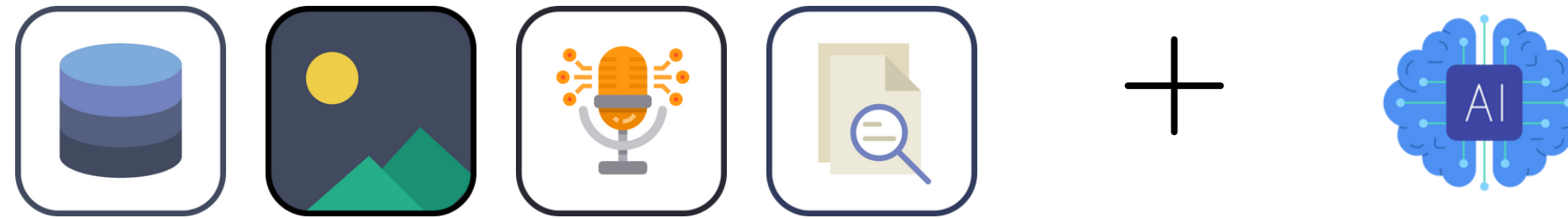


FlockMTL
Website

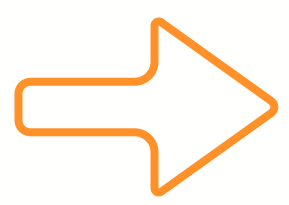
Problem Statement

DATA

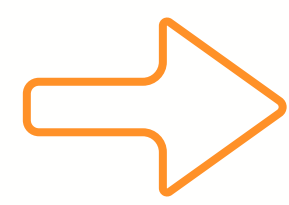
LLMs



- 1
- 2
- 3
- 4
- 5
- 6
- 7



FlockMTL

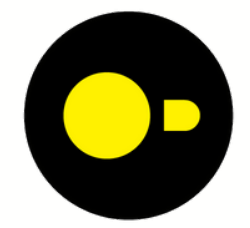


LLMs

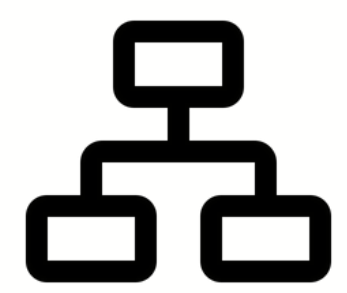


FlockMTL Website

Our Solution



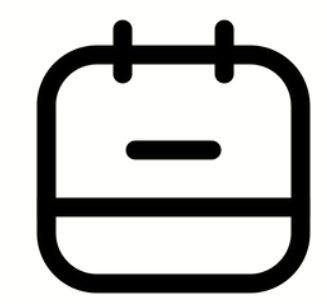
DuckDB



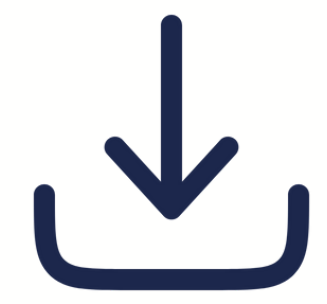
2 New First-Class Schema Objects



12 Scalar & Aggregate Functions



500+ Weekly Downloads



11K+ Downloads Total



FlockMTL
Website

Resource Independence

1

2

3

4

5

6

7

```
CREATE MODEL(  
  'model_name',  
  'model',  
  'provider',  
  {  
    "tuple_format": "JSON",  
    "batch_size": 8,  
    "model_parameters": {  
      "temperature": 0.2,  
      "top_p": 0.95  
    }  
  }  
);
```

Model Creation

```
CREATE PROMPT(  
  'prompt_name',  
  'prompt'  
);
```

Prompt Creation



FlockMTL
Website

FlockMTL Functions

Function	Description
<code>llm_complete</code>	generates text or structured output from an input tuple or an image.
<code>llm_filter</code>	returns True/False for a given tuple or an image.
<code>llm_embedding</code>	generates an embedding vector from an input text value.
<code>fusion</code>	fuses N normalized scores, e.g., using <code>rrf</code> / <code>combanz</code> / <code>combmed</code> / etc.

Scalar Functions

Function	Description
<code>llm_reduce</code>	generates text or structured output from multiple input tuples or images.
<code>llm_rerank</code>	ranks input tuples based on relevance.
<code>llm_first/last</code>	returns the most or least relevant tuple from multiple input tuples.

Aggregate Functions

1

2

3

4

5

6

7



FlockMTL
Website

Query Example

1

2

3

4

5

6

7

```
SELECT pi.*, qc.*
FROM product_images AS pi
JOIN qa_checklist AS qc
  ON pi.product_id = qc.product_id
WHERE llm_filter(
  { 'model_name': 'gpt-4o' },
  {
    'prompt': 'Based on image and text, does this example match the
described condition?',
    'context_columns': [
      { 'data': pi.image_data, 'type': 'image' },
      { 'data': pi.qa_description, 'name': 'qa_description' },
      { 'data': qc.checklist_text, 'name': 'checklist_text' }
    ]
  }
);
```

Semantic QA-Image Consistency Filter for Product
Quality Control

```
WITH QueryEmbedding AS (
  SELECT llm_embedding({'model':'text-embedding-3-small'},
    {'context_columns':{'data':'join algorithms in databases'}})::DOUBLE[1536] AS embedding
),
VS AS (
  SELECT rp.idx, rp.content,
    array_cosine_similarity(qe.embedding,
      llm_embedding({'model':'text-embedding-3-small'},
        {'context_columns':{'data':rp.content}})::DOUBLE[1536]) AS score
  FROM research_passages rp, QueryEmbedding qe
  ORDER BY score DESC LIMIT 100
),
BM25 AS (
  SELECT rp.idx, rp.content,
    fts_main_research_chunks.match_bm25(index_column,'join algorithms in databases', fields:='content') AS score
  FROM research_passages rp
  ORDER BY score DESC LIMIT 100
),
Top10 AS (
  SELECT bm.content AS doc1, vs.content AS doc2,
    fusion(bm.score::DOUBLE / (MAX(bm.score) OVER ()),
      vs.score::DOUBLE / (MAX(vs.score) OVER ())) AS fused_score
  FROM BM25 bm FULL OUTER JOIN VS vs ON bm.idx = vs.idx
  ORDER BY fused_score DESC LIMIT 10
)
SELECT llm_rerank({'model':'gpt-4o'},
  {'prompt':'mentions cyclic joins',
    'context_columns':{'data':doc1,'name':'document 1'},
    {'data':doc2,'name':'document 2'}}) AS rerank_result
FROM Top10;
```

Hybrid Search and Rerank for Join Algorithm Research Passages



FlockMTL
Website

Why DuckDB?



Portable



Powerful



EASY

Simple

- 1
- 2
- 3
- 4
- 5
- 6
- 7



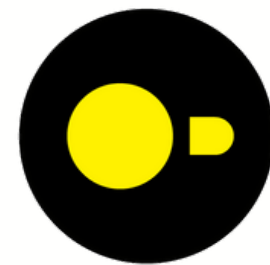
FlockMTL
Website



FlockMTL

+

=



DuckDB



Thank You!

1

2

3

4

5

6

7